สุทธิศักดิ์ พงศ์ธนาพาณิช

Advanced BASIC Programming Technique ตอน แนะนำ Visual Basic 5.0

คุณสมบัติและความสามารถใหม่ ๆ ของ Visual Basic 5.0

สำหรับในบทความตอนที่ผ่านมาเร็ว ๆ นี้ ผู้เขียนได้ทำการเขียนโปรแกรมตัวอย่างทั้งหมดด้วย Visual Basic 5.0 ซึ่งก็มีผู้อ่านบางท่านต้องการให้มีการเขียนบทความเพื่อแนะนำคุณสมบัติและความสามารถใหม่ ๆ ทั้งหมดของ Visual Basic 5.0 โดยสำหรับตัวผู้เขียนก็เห็นว่าเป็นการดีที่จะเขียนบทความดังกล่าว เพราะไมโครซอฟท์ได้ทำการปรับปรุง Visual Basic 5.0 จากเวอร์ชันที่ผ่านมาแบบรื้อโครงสร้างกันเลยทีเดียว ไม่ว่าจะเป็นเอนไวรอนเมนท์สำหรับการเขียนและทดสอบโค้ด คอมไพเลอร์ที่สามารถสร้างโค้ดเนทีฟ (Native code) คอนโทรลภายใต้สถาปัตยกรรม ActiveX และ COM เป็นต้น

ดังนั้นในบทความตอนนี้ผู้เขียนก็จะทำการอธิบายถึงคุณสมบัติและความสามารถใหม่ ๆ ทั้งหมดของ Visual Basic 5.0 โดยผู้เขียนได้ทำการคัดลอกข้อมูลบางส่วนมาจากหนังสือ "การใช้โปรแกรม Visual Basic 5.0 Professional" ซึ่งเป็นหนังสือที่ปรับปรุงใหม่ทั้งหมดจากหนังสือ "การใช้โปรแกรม Visual Basic 4.0 Professional" แต่งโดยตัวผู้เขียน ซึ่งคาดว่าจะออกจำหน่ายประมาณเดือนกรกฎาคม 2541 นี้ ซึ่งเนื้อหาทั้งหมดจะครอบคลุมวิธีการใช้งานคอนโทรล ActiveX ทั้งหมดของ Visual Basic 5.0 Professional พร้อมทั้งโปรแกรมตัวอย่างประกอบอย่างสมบูรณ์

คุณสมบัติใหม่ของ Visual Basic 5.0

บริษัทไมโครซอฟท์ได้มีการปรับปรุงโครงสร้างของ Visual Basic 5.0 ทั้งด้านของขีดความสามารถและเทคโนโลยีใหม่ ๆ ด้าน ActiveX ซึ่งได้ถูกกำหนดเป็นมาตรฐานด้านเทคโนโลยีในปัจจุบันไปเสียแล้ว และได้รวบรวมสิ่งเหล่านี้เอาไว้ใน Visual Basic 5.0 โดยมีการประกาศตัวในช่วงปี พ.ศ. 2540 ที่ผ่านมา แต่เนื่องจาก Visual Basic 5.0 สนับสนุนเฉพาะการสร้างโค้ดระดับ 32 บิต ภายใต้ระบบปฏิบัติการวินโดวส์ 95 และ NT 4.0 เท่านั้น ดังนั้นจึงไม่สามารถที่จะใช้งาน Visual Basic 5.0 กับวินโดวส์ 3.x สำหรับขีดความสามารถใหม่ของ Visual Basic 5.0 ที่ไมโครซอฟท์ได้ประกาศมีดังต่อไปนี้

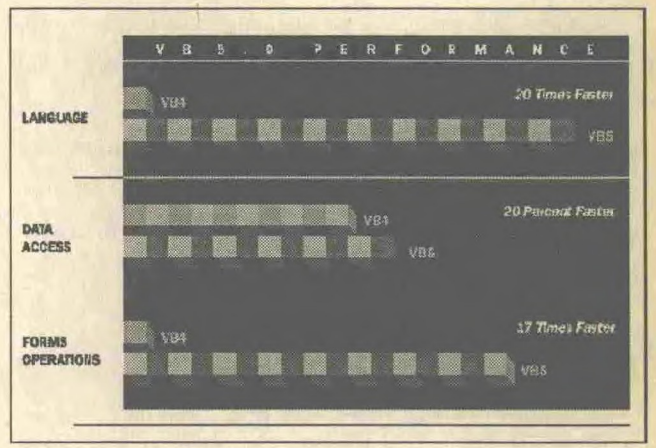
คอมไพแอพพลิเคชันในรูปแบบโค้ดแท้

Visual Basic 5.0 Professional หรือ Enterprise edition สนับสนุนการสร้างแอพพลิเคชัน (.exe) ทั้งในรูปแบบของโค้ดเทียม (p-code) หรือโค้ดแท้ (native code) โดยที่การคอมไพล์โปรเจ็กต์เพื่อสร้างไฟล์ .exe ในรูปแบบ

โค้ดแท้นั้น ผู้อ่านสามารถที่จะทำการออปติไมซ์โค้ดในรูปแบบของ .exe ทั้งในด้านของความเร็วและขนาดของไฟล์ .exe ได้ในหลาย ๆ เงื่อนไขอีกด้วย และนอกจากนี้ผู้อ่านยังสามารถที่จะเลือกสร้างโค้ด .exe ที่เหมาะสำหรับการทำงานกับไมโครโปรเซสเซอร์ Pentium PRO ของอินเทลเป็นการเฉพาะได้อีกด้วย เพราะจะทำให้โค้ดที่ได้สามารถโหลดและเข้าถึงข้อมูลได้เร็วที่สุด

โค้ดเทียม เป็นโค้ดที่โปรเซสเซอร์ไม่สามารถที่จะนำไปปฏิบัติการได้โดยตรง เพราะเป็นโค้ดที่ถูกออกแบบเป็นการเฉพาะเพื่อช่วยลดขนาดของไฟล์ .exe ที่ได้จากการคอมไพล์ ดังนั้นในขณะที่มีการเรียกใช้แอพพลิเคชันในรูปแบบ p-code นั้น จะมีโค้ดส่วนหนึ่งในไฟล์ .exe ที่ทำหน้าที่แปลงโค้ด p-code ให้เป็นโค้ดแท้ที่โปรเซสเซอร์เข้าใจและสามารถประมวลผลได้ทันทีต่อไป

โค้ดแท้ เป็นโค้ดที่โปรเซสเซอร์สามารถนำไปประมวลผลได้ทันที ดังนั้นจึงทำให้แอพพลิเคชันสามารถทำงานได้รวดเร็วกว่าโค้ดเทียมเป็นอย่างมาก แต่ก็มีข้อเสียตรงที่ไฟล์ .exe ที่ได้จากการคอมไพล์เป็นโค้ดแท้ จะมีขนาดใหญ่กว่าขนาดไฟล์ .exe ที่ได้จากการคอมไพล์เป็นเทียม สำหรับ Visual Basic 5.0 ผู้อ่านสามารถสร้างแอพพลิเคชันในรูปแบบโค้ดแท้ที่ทำงานได้เร็วกว่าแอพพลิเคชันในรูปแบบโค้ดเทียมถึง 20 เปอร์เซ็นต์ โดยเฉพาะโค้ดด้านการคำนวณคณิตศาสตร์จะสามารถทำงานได้เร็วกว่าโค้ดเทียมถึง 20 เท่าตัวเลยทีเดียว โดยในรูปที่ 1 แสดงผลการเปรียบเทียบความเร็วของ Visual Basic เวอร์ชัน 4.0 และ 5.0 ซึ่งถูกทำการเปรียบเทียบและตีพิมพ์โดยไมโครซอฟท์



รูปที่ 1 แสดงผลการเปรียบเทียบความเร็วของ Visual Basic เวอร์ชัน 4.0 และ 5.0

สร้างคอนโทรล ActiveX

Visual Basic 4.0 ไม่สนับสนุนการสร้างไฟล์คอนโทรล .ocx ทั้งนี้เนื่องจาก Visual Basic 4.0 ไม่สนับสนุนการสร้างโค้ดแท้นั่นเอง แต่สำหรับ Visual Basic 5.0 นั้น ผู้อ่านสามารถที่จะสร้างคอนโทรล .ocx ด้วยเทคโนโลยี ActiveX ได้โดยตรง โดยไม่จำเป็นต้องอาศัย Visual C++ อีกต่อไป ซึ่งจะทำให้ผู้อ่านสามารถสร้างคอนโทรล .ocx ได้อย่างหลากหลาย โดยสามารถที่จะรวมเอาคอนโทรลมาตรฐาน เช่น TextBox หรือ PictureBox เป็นต้น ของ Visual Basic มาใช้ในการสร้างคอนโทรล .ocx ของตนเองได้โดยตรง

การทำงานแบบหลายโปรเจ็กต์

เนื่องจาก Visual Basic 5.0 สนับสนุนการสร้างคอนโทรล .ocx ดังนั้นในขั้นตอนการทดสอบโค้ดของคอนโทรล .ocx นั้น ผู้อ่านต้องดำเนินการเช่นเดียวกับการใช้คอนโทรลในโปรเจ็กต์ทั่ว ๆ ไป โดยการเพิ่มคอนโทรลดังกล่าวลงในฟอร์มของโปรเจ็กต์อื่น ๆ เพราะคอนโทรล .ocx จะไม่สามารถทำงานได้โดยลำพัง ดังนั้นเพื่อรองรับการทดสอบคอนโทรล .ocx ที่ถูกสร้างขึ้นมา Visual Basic 5.0 จึงสนับสนุนการทำงานในลักษณะของหลาย ๆ โปรเจ็กต์

สำหรับความสามารถที่สำคัญของ Visual Basic 5.0 อีกประการก็คือสนับสนุนการพัฒนาแอพพลิเคชันในรูปแบบ Client-Server (ในปัจจุบันโปรแกรมเมอร์นิยมที่จะพัฒนาแอพพลิเคชันสำหรับระบบเครือข่ายตามแนวทางของเทคโนโลยี Three-Tiered Client-Server ที่จะมีแอพพลิเคชันทำงานที่เซอร์ฟเวอร์ระดับกลาง ซึ่งช่วยในการสื่อสารและกรองข้อมูลระหว่างแอพพลิเคชันไคลเอ็นต์และเซอร์ฟเวอร์) ดังนั้นจึงสามารถที่จะทดสอบแอพพลิเคชันที่สร้างขึ้นมาภายใต้เอนไวรอนเมนท์ของ Visual Basic 5.0 ได้อย่างสมบูรณ์

สร้างแอพพลิเคชันแบบ Three-Tiered

ผู้อ่านสามารถที่จะสร้างแอพพลิเคชันแบบ Three-Tiered ด้วย Visual Basic 5.0 โดยอาศัยสถาปัตยกรรมเทคโนโลยีแบบ DCOM (Distributed Component Object Model) สำหรับแอพพลิเคชันด้านไคลเอ็นต์แบบ 32 บิต ซึ่งสามารถที่จะทำงานภายใต้ระบบเครือข่าย LAN (Local Area Network) WAN (Wide Area Network) หรืออินเทอร์เน็ตได้อย่างมีประสิทธิภาพ

สร้างอินเทอร์เฟสแบบ SDI หรือ MDI

ผู้อ่านสามารถที่จะสร้างแอพพลิเคชันทั้งในรูปแบบ SDI (Single-Document Interface) หรือ MDI (Multi-Document Interface) และนอกจากนี้ยังสามารถที่จะสร้างแอพพลิเคชันที่มีหน้าตาเหมือนกับ Windows Explorer ได้อีกด้วย ซึ่งการสร้างแอพพลิเคชันในรูปแบบ SDI หรือ MDI นั้น ก็ขึ้นกับความเหมาะสมของแอพพลิเคชันแต่ละประเภท สำหรับตัวอย่างของแอพพลิเคชันดังกล่าวก็จะมีหน้าตาดังเช่นในรูปที่ 2 ซึ่งแสดงถึงแอพพลิเคชันทั้งสองประเภท

นอกจากแอพพลิเคชันในรูปแบบ SDI และ MDI แล้ว แอพพลิเคชันที่มีอินเทอร์เฟสเหมือนกับ Windows Explorer ดังในรูปที่ 3 ก็ได้รับความนิยมเป็นอย่างมากในปัจจุบัน ซึ่งอินเทอร์เฟสของ Windows Explorer นั้นประกอบด้วยหน้าต่างเดียว และมีการแบ่งพื้นที่แสดงผลออกเป็นสองข้าง โดยข้างซ้ายจะใช้แสดงผลโครงสร้างในรูปแบบแผนภูมิต้นไม้ (tree structure) ส่วนด้านขวาจะใช้แสดงรายละเอียดของไฟล์ทั้งหมดที่ถูกเลือกจากไดเรคทอรีทางด้านซ้าย

Wizard

การสร้างแอพพลิเคชันหรือคอนโทรลด้วย Visual Basic 5.0 นั้นไม่ใช่สิ่งที่ยากอีกต่อไป เพราะ Visual Basic 5.0 ได้มี wizard ที่จะช่วยออกแบบหรือสร้างโครงร่าง (layout) ของแอพพลิเคชันหรือคอนโทรลให้โดยอัตโนมัติ สำหรับประเภทของ wizard ที่สนับสนุนมีดังต่อไปนี้

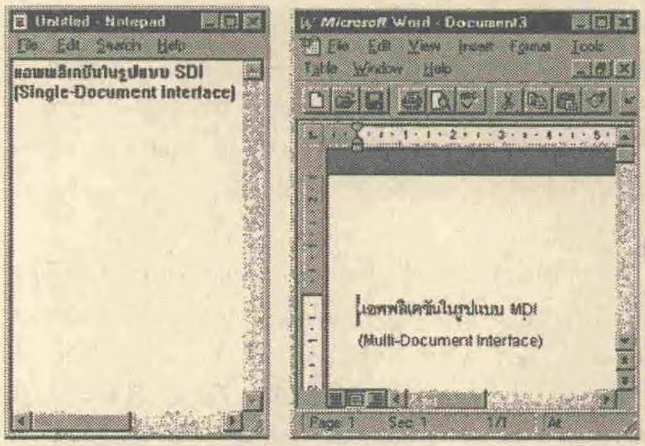
```
Application Wizard
ActiveX Control Interface Wizard
ActiveX Document Migration Wizard
Data Form Wizard
PropertyPage Wizard
Setup Wizard
Class Builder Wizard
Wizard Toolbar
Wizard Manager
```

Extensibility Model

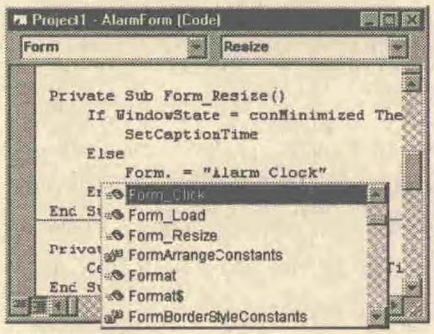
โมเดลเอนไวรอนเมนท์ของ Visual Basic 5.0 ที่ได้รับการพัฒนาให้มีขีดความสามารถและมีความยืดหยุ่นมากขึ้น ซึ่งผู้อ่านสามารถที่จะเพิ่มโปรแกรมชนิด Add-In เข้ากับเอนไวรอนเมนท์ เพื่อขยายขีดความสามารถของ Visual Basic 5.0 IDE ได้อย่างเต็มที่ ซึ่งช่วยให้ผู้อ่านสามารถที่จะปรับปรุงหรือตบแต่งเอนไวรอนเมนท์ในการพัฒนาซอร์สโค้ดได้ด้วยเทคโนโลยี IntelliSense ของ Visual Basic 5.0

สร้างโพรซีเดอร์เหตุการณ์ให้กับโมดูลคลาส

Visual Basic 4.0 สนับสนุนการสร้างออบเจ็กต์คลาสโดยอาศัยโมดูลคลาส แต่คลาสที่ถูกสร้างไม่สามารถที่จะสร้างเหตุการณ์เพื่อตอบสนองต่อการทำงาน ดังนั้นจึงทำให้ขาดความสมบูรณ์ในการติดต่อกับแอพพลิเคชันอื่น ๆ ในลักษณะของการสื่อสาร 2 ทาง (2 ways communication) แต่สำหรับ Visual Basic 5.0 ผู้อ่านสามารถที่จะสร้างโพรซีเดอร์เหตุการณ์ (event) ให้กับคลาสได้โดยตรง และสามารถที่จะเลือกเหตุการณ์ใด ๆ ให้มีการกระตุ้น (activate) ตามสถานการณ์ที่เหมาะสมได้เช่นกัน ซึ่งจะช่วยให้ออบเจ็กต์คลาสที่ได้มีความสมบูรณ์และถูกต้องตามแนวความคิดของการโปรแกรมเชิงวัตถุหรือ OOP (Object Oriented Programming) มากยิ่งขึ้น และนอกจากนี้ยังสนับสนุนการเขียนโปรแกรมสำหรับระบบเครือข่ายด้วยออบเจ็กต์คลาสอีกด้วย



รูปที่ 2 แอพพลิเคชันในรูปแบบ SDI และ MDI



รูปที่ 3 แสดงความสามารถด้านการแนะนำโค้ดคำสั่งของ code editor

Smart Code Editor

เอนไวรอนเมนท์ของ Visual Basic 5.0 ได้รับการปรับปรุงใหม่เป็นอย่างมาก โดยเฉพาะความสามารถของ code editor เช่น การออกแบบให้มีความคล้ายคลึงกับ word processor เช่น การเพิ่มปุ่มสำหรับเลือกรูปแบบการแสดงรูทีน และนอกจากนี้ยังมีสามารถในการแสดงคุณสมบัติ วิธี และเหตุการณ์ของฟอร์ม คอนโทรลหรือออบเจ็กต์ในขณะที่กำลังแก้ไขโค้ดส่วนที่เกี่ยวข้องได้อีกด้วย ซึ่งจะช่วยให้ผู้อ่านไม่ต้องจดจำคำสั่ง หรือความสามารถต่าง ๆ ของฟอร์ม คอนโทรล หรือออบเจ็กต์อีกต่อไป

Multithreading ActiveX Components

ActiveX component ที่ถูกสร้างด้วย Visual Basic 5.0 ไม่ว่าจะเป็น .exe .dll หรือ .ocx ก็ตาม ล้วนแต่สนับสนุนการทำงานในระบบมัลติทาสกิงแบบ preemtive ของวินโดวส์ 95 โดยที่แต่ละส่วนของงานหรือทริด (thread) ที่กำลังปฏิบัติงาน จะมีการควบคุมลำดับขั้นตอนของการปฏิบัติงานด้วยวิธี serializing requests (คำสั่งจะถูกจัดลำดับการทำงานอย่างเป็นขั้นตอนอย่างต่อเนื่อง ระหว่างที่มีการสลับไป-มาระหว่างแอพพลิเคชัน) ซึ่งเป็นคุณสมบัติที่สำคัญประการหนึ่งของระบบ Automation

เพราะจะช่วยป้องกันการซ้อนทับ หรือความไม่เป็นไปตามลำดับการทำงานของแต่ละส่วนของงานของแต่ละแอพพลิเคชันได้อย่างสมบูรณ์

ตัวปฏิบัติการ AddressOf

ฟังก์ชันวินโดว์ API บางฟังก์ชันนั้นต้องการการเรียกกลับ (call back) เช่น กลุ่มฟังก์ชัน Enum... เป็นต้น ระหว่างฟังก์ชันกับโค้ดที่เรียกใช้ฟังก์ชัน ซึ่งผู้อ่านจะไม่สามารถใช้ฟังก์ชันกลุ่มนี้กับ Visual Basic 4.0 ได้เลย เพราะว่า Visual Basic 4.0 ไม่สนับสนุนการส่งผ่านหรืออ้างอิงไปยังแอดเดรสของโพรซีเดอร์ในขณะที่กำลังปฏิบัติการ แต่สำหรับ Visual Basic 5.0 นั้นได้มีการเพิ่มความสามารถในการอ้างแอดเดรสของโพรซีเดอร์ในส่วนที่ต้องการ ให้สามารถเรียกกลับได้จากฟังก์ชันวินโดว์ API โดยใช้ตัวปฏิบัติการ AddressOf แต่มีข้อจำกัดที่โพรซีเดอร์ดังกล่าวจะต้องถูกสร้างในโมดูลมาตรฐาน (.bas) เท่านั้น ซึ่งผู้เขียนจะกล่าวรายละเอียดในบทที่เกี่ยวกับฟังก์ชันวินโดว์ API ต่อไป

Polymorphism

Polymorphism เป็นคำศัพท์ที่ใช้กับมากในการเขียนโปรแกรมตามแนวคิดของ Object Oriented Programming (OOP) ซึ่งหมายถึงการที่คลาสต่าง ๆ สามารถที่จะมีคุณสมบัติ หรือวิธีที่เหมือนกันได้ โดยที่ในการเรียกใช้คุณสมบัติ หรือวิธีเหล่านี้ โค้ดส่วนที่เรียกใช้ไม่จำเป็นต้องทราบถึงคลาสที่เป็นเจ้าของคุณสมบัติหรือวิธีเหล่านี้ล่วงหน้า สำหรับ Visual Basic 5.0 จะสนับสนุนความสามารถด้าน polymorphism นี้โดยอาศัยหลาย ๆ อินเทอร์เฟสของ ActiveX (อินเทอร์เฟสของ ActiveX หมายถึง กลุ่มของคุณสมบัติหรือวิธีที่สัมพันธ์กัน) ซึ่งเป็นความสามารถที่มีโครงสร้างมาจากเทคโนโลยีด้าน OOP ของไมโครซอฟท์ที่ชื่อ Component Object Model (COM)

ข้อมูลชนิด Enum

Enum เป็นการสร้างชุดของข้อมูลชนิดค่าคงที่เลขจำนวนเต็ม โดยที่ผู้อ่านสามารถสร้างตัวแปรให้เป็นข้อมูลชนิด Enum ได้ โดยที่ตัวแปรที่ถูกสร้างขึ้นมาก็สามารถมีค่าได้ตามค่าคงที่ ซึ่งถูกกำหนดเอาไว้ในข้อมูลชนิด Enum เช่น

```
Public Enum Days
   Sunday
   Monday
   Tuesday
   Wednesday
   Thursday
   Friday
   Saturday
End Enum
```

เป็นการสร้างข้อมูล Enum ชื่อ Days ซึ่งสามารถมีค่าได้ตั้งแต่ Sunday (มีค่าเท่ากับ 0) ถึง Saturday (มีค่าเท่ากับ 7) เป็นต้น ซึ่งข้อมูล Enum จะช่วยประหยัดพื้นที่ในการจัดเก็บเพราะในขณะที่โปรแกรมทำงานแต่ละสมาชิกของข้อมูล Enum จะใช้พื้นที่หน่วยความจำร่วมกันนั่นเอง

OLE Drag and Drop

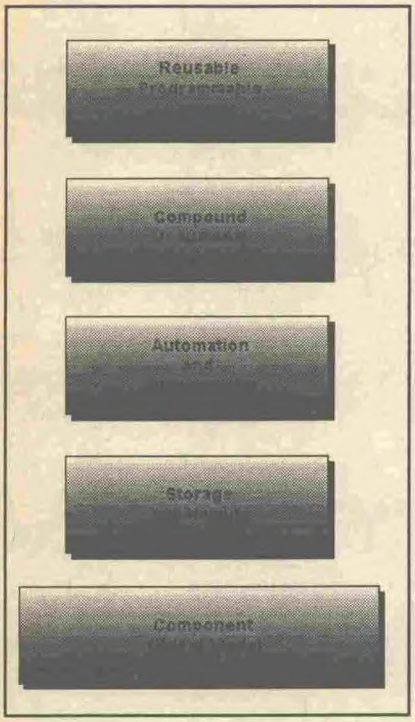
คอนโทรล .ocx ของ Visual Basic 5.0 โดยส่วนใหญ่สนับสนุนการทำงานแบบลากและวางของ OLE (OLE drag and drop, DnD) เช่น คอนโทรล PictureBox หรือ TextBox เป็นต้น ซึ่งผู้อ่านสามารถที่จะกำหนดให้คอนโทรลสนับสนุนการทำงานแบบ DnD ของ OLE ได้โดยการกำหนดให้คุณสมบัติ OLEDragMode และ OLEDropMode ของคอนโทรลให้มีค่าเท่ากับ 1 (vbOLEDragAutomatic) เท่านั้น อย่างเช่นในกรณีที่ผู้อ่านต้องการที่จะเปิดไฟล์ข้อความ .txt ภายในคอนโทรล TextBox ในขณะทำงานก็สามารถใช้ความสามารถ DnD ของ OLE โดยการลากไฟล์ข้อความ .txt มาวางลงในคอนโทรล TextBox เท่านั้น ซึ่งจะช่วยให้เกิดความสะดวกในการใช้งานเป็นอย่างมาก

Internet Programming

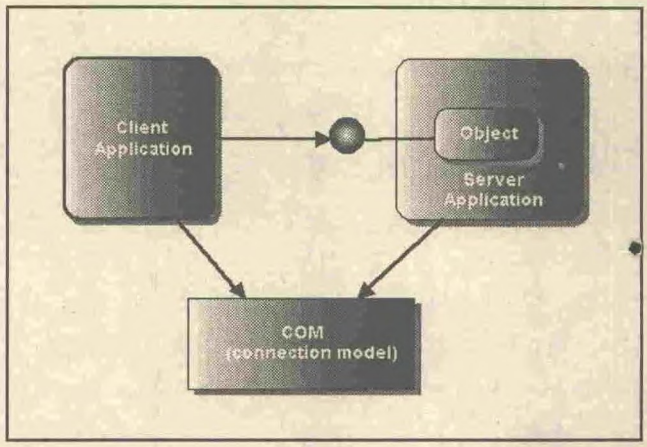
Visual Basic 5.0 สนับสนุนคอนโทรลที่ทำหน้าที่ติดต่อกับอินเทอร์เน็ต โดยอาศัยโปรโตคอล TCP/IP และนอกจากนี้ยังสนับสนุนความสามารถด้านการดาวน์โหลด internet component อีกด้วย ซึ่งเป็นการทำงานร่วมกับโปรแกรมประเภท browser เช่น Internet Explorer เป็นต้น ซึ่งจะมีการดาวน์โหลด internet component จากเครือข่ายอินเทอร์เน็ต เฉพาะเมื่อมีการอ้างอิงถึง component เหล่านั้นในหน้าของ HTML หรือเมื่อต้องการอัพเดต component เท่านั้น โดยวิธีนี้จึงทำให้ผู้อ่านไม่จำเป็นต้องการติดตั้ง component เหล่านี้ทั้งหมดลงในเครื่องคอมพิวเตอร์ เพราะจะเป็นหน้าที่ของ browser ที่จะคอยตรวจสอบและดาวน์โหลดให้เฉพาะเท่าที่จำเป็น และในปัจจุบันบรรดาบริษัทที่เป็นผู้ผลิตไฟล์คอนโทรล .ocx สำหรับ Visual Basic 5.0 ก็ได้ทำการพัฒนาคอนโทรลสำหรับอินเทอร์เน็ตกันอย่างมากมาย ซึ่งผู้อ่านสามารถค้นหารายละเอียดเกี่ยวกับคอนโทรล .ocx สำหรับ Visual Basic 5.0 ได้ทางอินเทอร์เน็ต

enhanced Data Access Objects

การสร้างโปรแกรมสำหรับการทำงานด้านฐานข้อมูลด้วย Visual Basic 5.0 ผู้อ่านสามารถที่จะใช้คุณสมบัติด้าน DAO (enhanced Data Access) เพื่อที่จะทำการเข้าถึงฐานข้อมูลในรูปแบบของ Microsoft Access (.mdb) ซึ่งเป็นฐานข้อมูลรูปแบบหลักของ Visual Basic 5.0 ได้อย่างรวดเร็ว ซึ่งผู้อ่านสามารถที่จะประมวลผลเรคอร์ดของฐานข้อมูลได้ทั้งวิธีการเข้าถึงครั้งละเรคอร์ดหรือวิธี SQL ก็ได้



รูปที่ 4
แสดงบริการแอพพลิเคชัน OLE ระดับสูงที่ถูกสร้างลงใน COM



รูปที่ 5 แสดงการติดต่อระหว่างไคลเอ็นต์กับเซอร์ฟเวอร์ภายใต้โมเดล COM

จากคุณสมบัติใหม่ ๆ ข้างต้นของ Visual Basic 5.0 ผู้อ่านจะเห็นว่าเป็นคุณสมบัติที่เกี่ยวข้องกับเทคโนโลยีใหม่ที่เรียกกันว่า ActiveX และ COM แต่เนื่องจากในหนังสือเล่มนี้ผู้เขียนจะมุ่งเน้นเฉพาะการใช้งานคอนโทรล ActiveX ทั้งหมดของ Visual Basic 5.0 เท่านั้น ดังนั้นก็จะไม่ขอกล่าวลึกลงไปถึงการการสร้างองค์ประกอบด้าน ActiveX เช่น .exe, .dll หรือ .ocx แต่เพื่อเป็นการเสริมความเข้าใจเกี่ยวกับเทคโนโลยีใหม่ของไมโครซอฟท์ ผู้เขียนก็จะขออธิบายคร่าว ๆ ถึงพื้นฐานของเทคโนโลยี ActiveX และ COM เพื่อเป็นเกร็ดความรู้ในหัวข้อต่อไป

COM

COM (Component Object Model) เป็นสถาปัตยกรรมซอฟต์แวร์ที่ได้กำหนดมาตรฐาน สำหรับช่วยให้องค์ประกอบ (component) ที่ถูกสร้างโดยผู้ผลิตซอฟต์แวร์รายต่าง ๆ สามารถที่จะใช้งานร่วมกับแอพพลิเคชันต่าง ๆ ได้ (interoperability) โดยจะไม่ขึ้นกับภาษาที่ใช้ในการพัฒนาองค์ประกอบ ซึ่งสถาปัตยกรรมพื้นฐานของ COM จะสนับสนุนการให้บริการซอฟต์แวร์ระดับสูง (high-level software) เช่นเดียวกับ OLE ไมโครซอฟท์ได้มีการประยุกต์สถาปัตยกรรม COM นี้ให้กับหลาย ๆ ออบเจ็กต์ เช่น คอนโทรล, compound document, automation, data transfer หรือ storage และ naming เป็นต้น

สำหรับแนวคิดพื้นฐานหรือมาตรฐานที่ได้ถูกกำหนดเอาไว้ใน COM มีดังต่อไปนี้

- มาตรฐานไบนารีสำหรับการเรียกใช้ฟังก์ชันระหว่างองค์ประกอบ
- การเตรียมการรวมกลุ่ม strongly-typed ของฟังก์ชันเข้ามายังอินเทอร์เฟส ซึ่งอินเทอร์เฟสพื้นฐานต้องมีการเตรียมวิธีสำหรับองค์ประกอบใช้ในการค้นหาอินเทอร์เฟส ที่ถูกจัดการโดยองค์ประกอบอื่น ๆ และวิธีการนับจำนวนการอ้างอิงถึง (component referenced) เพื่อที่จะทราบถึงอายุ (lifetime) ขององค์ประกอบนั้น ๆ
- กลไกการชี้ระบุองค์ประกอบและอินเทอร์เฟสขององค์ประกอบ
- ต้องมี component loader ที่ช่วยติดตั้งองค์ประกอบในกรณีการติดต่อข้ามเครือข่าย เพื่อช่วยในการจัดการด้านการติดต่อระหว่างองค์ประกอบ

COM เป็นโมเดลที่ถูกออกแบบภายใต้เงื่อนไขของการเขียนประเภทโปรแกรมเชิงวัตถุ (object-based programming) เพื่อช่วยให้การติดต่อระหว่างแอพพลิเคชันเป็นไปโดยไม่จำเป็นต้องมีการติดต่อผ่านตัวกลาง เพราะจะทำให้เกิดความล่าช้าในการทำงานหรือส่งผ่านข้อมูลนั่นเอง

ActiveX

ActiveX เป็นเทคโนโลยีที่ถูกพัฒนาโดยไมโครซอฟท์ เพื่อช่วยให้สามารถโต้ตอบระหว่างผู้ใช้กับเครือข่าย WWW (World Wide Web) ของอินเทอร์เน็ต ไม่ว่าจะเป็นด้านมัลติมีเดีย ออบเจ็กต์ที่สามารถโต้ตอบได้ และนอกจากนี้ ActiveX ยังช่วยให้แอพพลิเคชันที่ได้ สามารถทำงานภายใต้แพลตฟอร์มหรือระบบปฏิบัติการที่ต่างกัน เช่น วินโดวส์ ยูนิกซ์ (UNIX) เป็นต้น สำหรับ Visual Basic 5.0 จะช่วยให้ผู้อ่านสามารถสร้างองค์ประกอบทั้งไคลเอ็นต์และเซอร์ฟเวอร์ ซึ่งองค์ประกอบที่สนับสนุนเทคโนโลยี ActiveX ของไมโครซอฟท์มีดังต่อไปนี้

- คอนโทรล ActiveX เป็นคอนโทรลที่สนับสนุนการโต้ตอบไป-มาภายใต้หน้าของ Web สำหรับ Visual Basic 5.0 คอนโทรล ActiveX

(.ocx) นั้นเป็นเครื่องมือที่ช่วยในการพัฒนาโปรแกรม เช่น ImageList หรือ Animation เป็นต้น ซึ่งมีชื่อเรียกอย่างเป็นทางการว่า OLE Control

- เอกสาร ActiveX (ActiveX Document) เป็นออบเจ็กต์ที่ช่วยในการขอดูเอกสารที่ไม่ได้อยู่ในรูปแบบ HTML ภายใต้โปรแกรม Web browser เช่น ไฟล์ของไมโครซอฟท์เวิร์ด หรือ ไมโครซอฟท์เอ็กเซล เป็นต้น ซึ่งมีชื่อเรียกอย่างเป็นทางการว่า DocObjects

- Active Scripting เป็นออบเจ็กต์ที่ช่วยรวมเอาคอนโทรล ActiveX และ แอพเพลตของจาวา (Java Applet) จาก browser

- Java Virtual Machine เป็นโค้ดที่ช่วยให้ browser เช่น Internet Explorer 3.x สามารถรันแอพเพลตของจาวาได้

- ActiveX Server Framework สนับสนุนฟังก์ชันสำหรับ Web Server ด้านต่าง ๆ เช่น การเข้าถึงฐานข้อมูล การรักษาความปลอดภัย เป็นต้น ซึ่งมีชื่อเรียกอย่างเป็นทางการว่า ISAPI

ActiveX เป็นเทคโนโลยีที่มีพื้นฐานด้านสถาปัตยกรรมซอฟต์แวร์ของมาตรฐาน COM ดังนั้นจึงสามารถที่จะใช้งานร่วมกันได้กับทุก ๆ แอพพลิเคชันหรือภาษาที่ถูกออกแบบตามมาตรฐาน COM และในปัจจุบัน ActiveX ซึ่งถูกจัดเป็นเทคโนโลยีรุ่นที่ 3 ที่สนับสนุนหลาย ๆ แพลตฟอร์ม ทั้งอินเทอร์เน็ตและแอพพลิเคชันที่ทำงานแบบไคลเอ็นต์เซอร์ฟเวอร์ภายใต้ระบบเครือข่ายอินทราเน็ต (intranet)

ความต้องการด้านฮาร์ดแวร์สำหรับ Visual Basic 5.0

สำหรับเครื่องคอมพิวเตอร์ที่จะสามารถทำงานร่วมกับ Visual Basic 5.0 ได้นั้น ไมโครซอฟท์ได้มีการกำหนดความต้องการขั้นต่ำของระบบที่เหมาะสมกับการทำงานของ Visual Basic 5.0 เอาไว้สำหรับการเลือกฮาร์ดแวร์ที่เหมาะสม ดังนี้

- เครื่องคอมพิวเตอร์ที่มีโปรเซสเซอร์ 486DX/66 MHz ขึ้นไป (แนะนำ Pentium ขึ้นไป)

- ระบบปฏิบัติการวินโดวส์ 95 หรือวินโดวส์ NT Workstation 4.0

- สำหรับระบบปฏิบัติการวินโดวส์ 95 แนะนำหน่วยความจำ (RAM) 12 เมกะไบต์ขึ้นไป แต่สำหรับวินโดวส์ NT Workstation 4.0 แนะนำหน่วยความจำ (RAM) 16 เมกะไบต์ขึ้นไป

- พื้นที่ว่างของฮาร์ดดิสก์

- สำหรับการติดตั้งแบบ Minimum ต้องการอย่างน้อย 30 เมกะไบต์

- สำหรับการติดตั้งแบบ Complete ต้องการอย่างน้อย 115 เมกะไบต์ (220 เมกะไบต์ ถ้ารวมเอกสารแบบ online ด้วย)

- ไดรฟ์ CD-ROM

- จอภาพ VGA หรือที่มีความละเอียดมากกว่า

- เมาส์

รู้จักกับเอนไวรอนเมนต์ของ Visual Basic 5.0

ผู้อ่านสามารถเริ่มต้นรัน Visual Basic ได้โดยการคลิกที่ปุ่มคำสั่ง Start และเลือกโปรแกรม Visual Basic 5.0 ในกรุ๊ป Microsoft Visual Basic 5.0 ก็จะปรากฏหน้าตาเอนไวรอนเมนท์ทั้งหมดของ Visual Basic ดังในรูปที่ 1 สำหรับ Visual Basic 5.0 ได้มีการออกแบบเอนไวรอนเมนต์ให้มีความแตกต่างจากเวอร์ชันที่ผ่านมาพอสมควร โดยจะมีการรวมเอาหน้าต่างทั้งหมดให้สามารถแสดงผลได้เฉพาะในขอบเขตของเอนไวรอนเมนต์เท่านั้น และที่สำคัญหน้าต่างทั้งหมดไม่ว่าจะเป็นหน้าต่างโปรเจ็กต์ (Project Explorer window) หน้าต่างคุณสมบัติ (Property window) หรือ แถบเครื่องมือ (Toolbar) ก็ล้วนแต่สามารถที่จะแสดงผลแบบหน้าต่างอิสระ หรืออย่างต่อเนื่อง (dockable) ก็ได้ และนอกจากยังอนุญาตให้ผู้อ่านสามารถที่จะทำการปรับแต่งแถบเครื่องมือตามความต้องการ ไม่ว่าจะเป็นการลบทิ้งหรือเพิ่มรายการแถบเครื่องมือที่มีความต้องการใช้งานบ่อย ๆ ให้สามารถแสดงผลในแถบเครื่องมือได้อีกด้วย

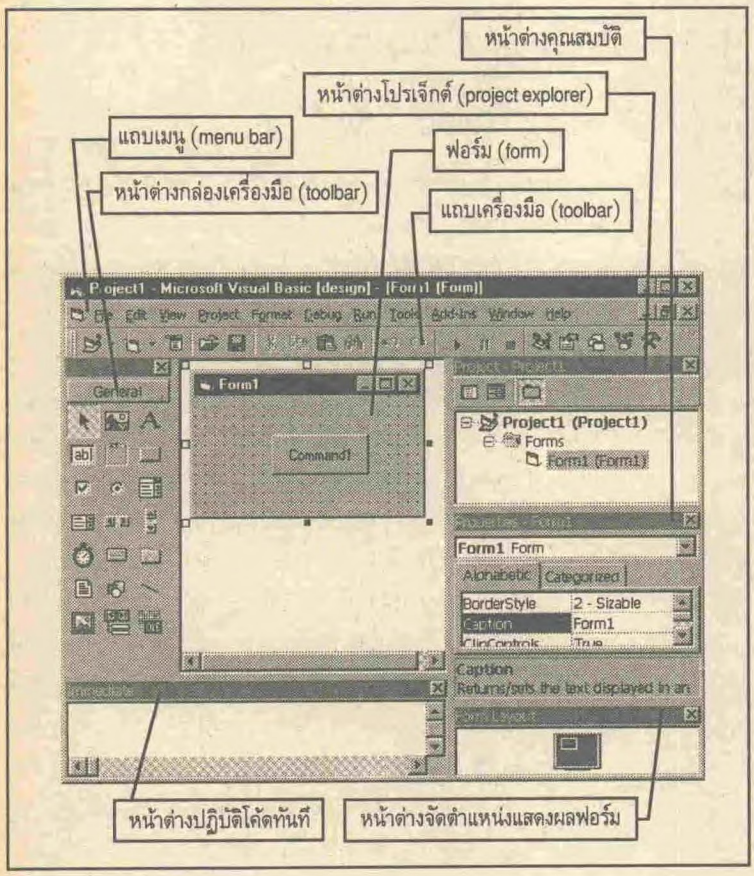
จากรูปที่ 6 ผู้อ่านจะเห็นว่าได้มีการออกแบบในส่วนของไอคอนสำหรับแสดงผลในแถบเครื่องมือ และหน้าต่างคอนโทรล (.ocx) ใหม่ทั้งหมด โดยจะแสดงผลทุก ๆ ไอคอนอย่างต่อเนื่อง โดยไม่มีการแบ่งแต่ละไอคอนออกเป็นปุ่ม ๆ แต่ทันทีที่ผู้อ่านเลื่อนเมาส์มาวางบนไอคอนแต่ละรูป ก็จะปรากฏกรอบสี่เหลี่ยมแสดงผลล้อมรอบไอคอนนั้น ๆ ทันที นอกจากนี้บางไอคอนของแถบเครื่องมือ เช่น ไอคอนของคำสั่ง Add Form ยังสามารถที่จะแสดงผลในรูปแบบของเมนู drop-down ได้อีกด้วย ซึ่งจะช่วยประหยัดจำนวนไอคอนที่ทำงานในลักษณะเดียวกันได้เป็นอย่างมาก และง่ายต่อการค้นหาและจดจำอีกด้วย

เนื่องจากเอนไวรอนเมนท์ของ Visual Basic 5.0 ได้รับการปรับปรุงให้ความสามารถมากขึ้น ดังนั้นถ้าหากผู้อ่านรันวินโดวส์ 95 บนเครื่องระดับ 80486 หรือ Pentium ที่มีหน่วยความจำน้อย ๆ (น้อยกว่า 16 เมกะไบต์) ก็อาจจะส่งผลต่อความเร็วในการตอบสนองของเอนไวรอนเมนต์ได้เช่นกัน เพราะวินโดวส์อาจจะต้องมีการติดต่อกับฮาร์ดดิสก์บ่อยครั้งขึ้น อันเนื่องมาจากกราฟิกที่ใช้ในการสร้างเอนไวรอนเมนต์ที่มากขึ้นนั่นเอง

จากรูปที่ 6 องค์ประกอบต่าง ๆ สำหรับเอนไวรอนเมนต์ของ Visual Basic มีหน้าที่และความหมายดังต่อไปนี้

แถบเมนู (menu bar)

เป็นแถบที่แสดงผลรายการเมนูหลักทั้งหมดของ Visual Basic ซึ่งจะแสดงผลใต้แถบชื่อเรื่อง (title bar) ของเอนไวรอนเมนต์ สำหรับเวอร์ชัน 5.0 จะประกอบด้วยเมนูหลักทั้งหมด 11 เมนู เช่น เมนู File, Edit หรือ Help เป็นต้น ดังในรูปที่ 6



รูปที่ 6
แสดงเอนไวรอนเมนท์
ของ Visual Basic 5.0

หน้าต่างโปรเจ็กต์ (project explorer window)

รวบรวมรายชื่อของกลุ่มของโปรเจ็กต์ (ในกรณีที่แก้ไขแอพพลิเคชันในรูปแบบของหลาย ๆ โปรเจ็กต์) ฟอร์ม โมดูลมาตรฐาน โมดูลคลาส (class module) ไฟล์ทรัพยากร (resource file) สำหรับการสร้างแอพพลิเคชัน ซึ่งการรวมเอาไฟล์เหล่านี้เข้าด้วยกันเพื่อสร้างแอพพลิเคชันภายใต้ Visual Basic เราเรียกว่า โปรเจ็กต์ (project) หรือโครงการ ซึ่งโดยปกติ Visual Basic จะจัดเก็บแต่ละไฟล์โปรเจ็กต์โดยมีนามสกุล .vbp (Visual Basic 2.0, 3.0 จะใช้นามสกุล .mak) โดยจะมีการจัดเก็บแยกออกจากไฟล์อื่น ๆ โดยเด็ดขาด

ฟอร์ม (form)

ฟอร์มเป็นหน้าต่างชนิดเดียวของ Visual Basic ที่ผู้อ่านสามารถในแสดงผลเพื่อที่จะสื่อการทำงานกับผู้ใช้ ทั้งนี้เนื่องจาก Visual Basic ได้รับการออกแบบให้โปรแกรมเมอร์สามารถนำมาใช้ออกแบบแอพพลิเคชันในลักษณะของการสื่อด้วยรูปภาพ (visual) ดังนั้นการสื่อการทำงานต่าง ๆ ระหว่างแอพพลิเคชันกับผู้ใช้ จะต้องกระทำผ่านฟอร์ม สำหรับเวอร์ชัน 5.0 ผู้อ่านสามารถสร้างแอพพลิเคชันมาตรฐานได้ 2 รูปแบบ คือ SDI (single document interface) และ MDI (multi-document interface) ดังนั้นผู้อ่านจึงสามารถเลือกใช้ได้ทั้งฟอร์มชนิด MDI หรือ FRM ก็ได้

สำหรับในโปรเจ็กต์หนึ่ง ๆ สามารถมีได้หลายฟอร์ม และภายในฟอร์มต่าง ๆ ก็จะถูกใช้ในการบรรจุคอนโทรลต่าง ๆ เช่น TextBox, PictureBox หรือ Image เป็นต้น ดังนั้นฟอร์มจึงทำหน้าที่เป็นตัวบรรจุ (container) และฟอร์มก็ยังเป็นออบเจ็กต์ตัวหนึ่งของ Visual Basic ที่อนุญาตให้ผู้อ่านแก้ไขคุณสมบัติได้ในขณะออกแบบจากหน้าต่างคุณสมบัติ และสามารถควบคุมพฤติกรรมต่าง ๆ ของฟอร์มได้โดยผ่านทางวิธี (method) เช่นเดียวกับออบเจ็กต์อื่น ๆ

หน้าต่างจัดตำแหน่งแสดงผลฟอร์ม (form layout window)

สำหรับเวอร์ชัน 5.0 แทนที่ผู้อ่านจะทำการเขียนโค้ดลงในโพรซีเดอร์เหตุการณ์ Form_Load หรือ Form_Activate เพื่อที่จะจัดตำแหน่งการแสดงผลฟอร์มบนจอภาพ เช่น โค้ดตัวอย่างการแสดงผลฟอร์ม Form1 ตรงกึ่งกลางของจอภาพดังนี้

```
Form1.Move (Screen.Width - Form1.ScaleWidth) /_
     2, (Screen.Height - Form1.ScaleHeight) / 2
```

แต่ถ้าหากผู้อ่านใช้คุณสมบัติใหม่ของ Visual Basic 5.0 ผู้อ่านสามารถที่จะใช้เมาส์ลากรูปฟอร์มในหน้าต่างจัดตำแหน่งแสดงผลฟอร์ม (ดูหน้าต่างมุมล่างขวาของรูปที่ 6) ไปยังตำแหน่งใด ๆ ในหน้าต่างก็ได้ เพื่อกำหนดตำแหน่งจริงในการแสดงผลฟอร์มนั้น ๆ บนจอภาพ

หน้าต่างคุณสมบัติ (property window)

รวบรวมคุณสมบัติทั้งหมดของฟอร์มหรือคอนโทรลเอาไว้ ซึ่งคุณสมบัติทั้งหมดที่ปรากฏในหน้าต่างนี้ จะเป็นคุณสมบัติที่ผู้อ่านสามารถกำหนดค่าได้ในขณะกำลังออกแบบ เช่น Caption, Text, Left, Top หรือ BackColor เป็นต้น เมื่อผู้อ่านทำการแก้ไขค่าคุณสมบัติต่าง ๆ ในหน้าต่างคุณสมบัตินี้ ก็จะส่งผลต่อออบเจ็กต์หรือคอนโทรลนั้น ๆ ทันที ซึ่งบางคุณสมบัติสามารถแสดงให้เห็นการเปลี่ยนแปลงคุณลักษณะได้ทันที เช่น Left, Top หรือ ForeColor เป็นต้น ส่วนคุณสมบัติบางอย่างจะสามารถแสดงผลได้ ก็ต่อเมื่อผู้อ่านมีการรันแอพพลิเคชันเท่านั้น เช่น Enabled หรือ Visible เป็นต้น สำหรับการแก้ไขคุณสมบัติของคอนโทรล ผู้อ่านสามารถแก้ไขได้ทั้งการเขียนโค้ดและจากหน้าต่างคุณสมบัติ

หน้าต่างปฏิบัติโค้ดทันที (immediate window)

เป็นหน้าต่างสำหรับใช้ในการทดสอบโค้ดโดยไม่จำเป็นต้องทำการคอมไพล์โปรเจ็กต์ เช่น สมมติว่าผู้อ่านต้องการทดสอบผลลัพธ์ของโค้ดดังนี้

```
Print Val("$123.3"), Val("123.3$")
```

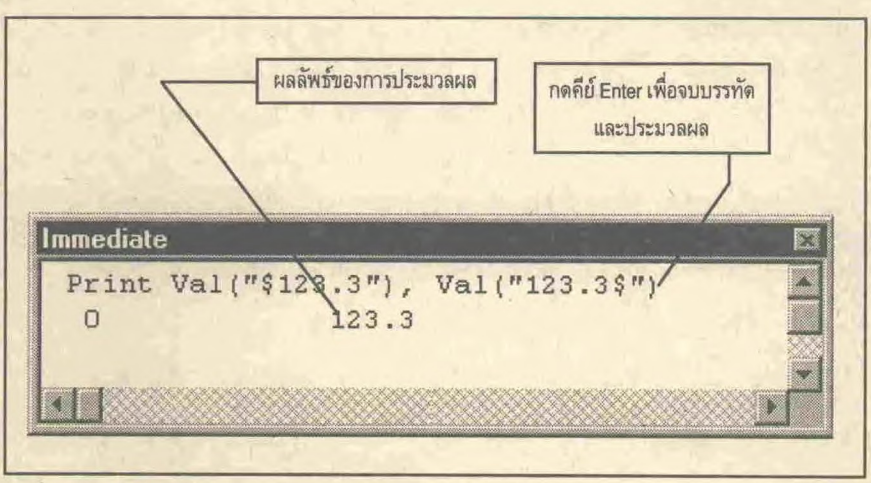
ผู้อ่านก็สามารถกระทำได้โดยการพิมพ์โค้ดบรรทัดดังกล่าวลงในพื้นที่บรรทัดใด ๆ ของหน้าต่างปฏิบัติโค้ดทันทีนี้ และกดคีย์ Enter เพื่อจบบรรทัด จากนั้น Visual Basic ก็จะทำการประมวลผลโค้ดบรรทัดดังกล่าว และรายงานผลลงในหน้าต่างปฏิบัติโค้ดทันที เช่นกัน

หน้าต่างกล่องเครื่องมือ (toolbox window)

เป็นหน้าต่างที่รวบรวมเอาเครื่องมือต่างๆ (คอนโทรล .ocx) สำหรับใช้ในขณะออกแบบฟอร์ม ดังนั้นผู้อ่านจึงสามารถเรียกชื่ออีกอย่างได้ว่า "ทูลบ็อกซ์" ซึ่งการรวมเอาเครื่องมือ (คอนโทรล) ตัวใดให้กับฟอร์ม ผู้อ่านสามารถที่ได้โดยการคลิกที่คอนโทรลนั้นเพื่อเลือกแล้วนำมาวางลงในฟอร์มโดยวิธีการลากแล้ววาง (DnD) หรือโดยวิธีการดับเบิลคลิกที่ไอคอนคอนโทรลในหน้าต่างกล่องเครื่องมือก็ได้ ซึ่ง Visual Basic ก็จะทำการวางคอนโทรลลงในฟอร์มปัจจุบันโดยอัตโนมัติ

แถบเครื่องมือ (toolbar)

เป็นแถบที่ปรากฏอยู่ด้านล่างของแถบเมนู ซึ่งเป็นส่วนที่รวบรวมไอคอนเล็ก ๆ มากมายเอาไว้ ซึ่งแต่ละไอคอนเหล่านี้ก็เปรียบเสมือนคำสั่งหนึ่ง ๆ ของเมนูหลัก ดังนั้นไอคอนในส่วนนี้จึงถูกออกแบบเพื่อให้การเลือกใช้คำสั่งของเมนูรวดเร็วและมีลักษณะที่สื่อความหมายกับผู้ใช้มากขึ้น โดยได้มีการเพิ่มเติมความสามารถด้านสื่อการใช้งานที่เรียกว่า Tool Tip โดยเมื่อผู้อ่านนำเมาส์พอยน์เตอร์ไปวางบนไอคอนใด ๆ คำสั่งที่เทียบเท่ากับไอคอนนั้น ก็จะปรากฏด้านล่างของเมาส์พอยน์เตอร์ทันที เช่น ซึ่งเทคนิคนี้ผู้อ่านบางท่านที่เคยใช้ไมโครซอฟต์เวิร์ดหรือแอพพลิเคชันในชุดไมโครซอฟต์ออฟฟิศ 97 คงจะคุ้นเคยกันดีอยู่แล้ว ส่วนไอคอนที่มีในทูลบาร์ทั้งหมดมีหน้าที่เทียบเท่ากับคำสั่งต่าง ๆ ในเมนูของ Visual Basic 5.0 มีดังต่อไปนี้



รูปที่ 7 แสดงตัวอย่างการแสดงผลของหน้าต่างปฏิบัติโค้ดทันที

คอนโทรล ActiveX ของ Visual Basic

คอนโทรล (.ocx) เป็นเครื่องมืออย่างหนึ่งที่ช่วยให้การสร้างแอพพลิเคชันด้วย Visual Basic สามารถกระทำได้ง่าย และรวดเร็วยิ่งขึ้น สำหรับ Visual Basic ได้มีการแบ่งคอนโทรลออกเป็น 2 กลุ่ม ดังนี้ คอนโทรลภายในมาตรฐาน (internal standard control) หรือที่เรียกอีกอย่างว่า intrinsic control เป็นคอนโทรลที่ไม่ได้เก็บแยกออกเป็นไฟล์ .ocx แต่เป็นคอนโทรลที่ถูกสร้างรวมเอาไว้ใน Visual Basic ซึ่งถูกเก็บเอาไว้ในไฟล์รันไทม์ไลบรารีของ Visual Basic (MSVBVM50.DLL) ดังนั้นทันทีที่ผู้อ่านโหลดเอนไวรอนเมนท์ของ Visual Basic ก็จะสามารถใช้งานคอนโทรลเหล่านี้ได้โดยอัตโนมัติ ส่วนอีกกลุ่มก็ได้แก่ คอนโทรล ActiveX (ActiveX control) เป็นคอนโทรลที่ถูกจัดเก็บแยกออกเป็นไฟล์ .ocx ซึ่งก่อนที่ผู้อ่านจะสามารถใช้งานคอนโทรลเหล่านี้ ผู้อ่านก็ต้องทำการเชื่อมต่อไฟล์ (.ocx) ที่เกี่ยวข้องเข้ากับเอนไวรอนเมนท์ของ Visual Basic เสียก่อนโดยการใช้คำสั่ง Components ดังที่จะได้กล่าวต่อไปในหัวข้อเมนู Project ในบทนี้ สำหรับไฟล์คอนโทรลในกลุ่มนี้จะถูกลงทะเบียนและติดตั้งลงในไดเรคทอรี "\Windows\System" ในขั้นตอนการติดตั้ง Visual Basic โดยอัตโนมัติ ซึ่งสามารถดูข้อมูลในไฟล์ลงทะเบียน (registry file) ได้โดยใช้โปรแกรม Registry Editor ของวินโดวส์

คอนโทรล ActiveX ที่ถูกเชื่อมต่อมายังเอนไวรอนเมนท์นั้น ก็จะถูกแสดงผลโดยเป็นไอคอนในหน้าต่างกล่องเครื่องมือ ดังในรูปที่ 8 และชื่อไฟล์ที่ถูกโหลดก็จะไม่แสดงผลในหน้าต่างโปรเจ็กต์ อย่างเวอร์ชัน 3.0 เนื่องจากคอนโทรลต่าง ๆ ตามที่กล่าวมาแล้วข้างต้นเป็นออบเจ็กต์ชนิดหนึ่งของ Visual Basic ดังนั้นทุก ๆ คอนโทรลก็จะต้องมีชื่อคลาส (class name) เป็นของตัวเอง และเพื่อป้องกันการสับสนของผู้อ่าน การเขียนถึงชื่อของคอนโทรลในหนังสือเล่มนี้ ผู้เขียนก็จะใช้ชื่อคลาสของคอนโทรลในการอ้างถึง เช่น คอนโทรล Picture Box ก็จะเรียกด้วยชื่อ PictureBox หรือคอนโทรล Multimedia MCI ก็จะเรียกด้วยชื่อ MMControl เป็นต้น

คอนโทรลมาตรฐานและคอนโทรล ActiveX แต่ละตัวในความเป็นจริงก็คือโค้ดชุดหนึ่ง (library) ที่สามารถทำงานได้เฉพาะตามที่กำหนด และตัวคอนโทรลก็มีคุณสมบัติ (property) เฉพาะตัวที่ผู้อ่านสามารถแก้ไขได้ และนอกจากนี้เมื่อมีการกระทำใด ๆ กับคอนโทรล เช่น การคลิก (click) การเปลี่ยนค่า (change) เป็นต้น เหตุการณ์ที่เกี่ยวข้องของคอนโทรลก็จะถูกเรียก (event-driven) ซึ่งผู้อ่านสามารถที่จะเขียนโค้ดเพื่อควบคุมการทำงานเมื่อเกิดเหตุการณ์ต่าง ๆ ได้โดยการเขียนโค้ดลงในส่วนโพรซีเยอร์เหตุการณ์ (event procedure)

เปรียบเทียบคุณสมบัติแต่ละรุ่นของ Visual Basic 5.0

Visual Basic 5.0 ได้ถูกแบ่งออกเป็น 3 รุ่น เพื่อให้เหมาะสมกับความต้องการและกำลังซื้อของกลุ่มผู้ใช้ ดังต่อไปนี้

Visual Basic 5.0 รุ่น Control Creation

เป็นรุ่นที่พื้นฐานที่สุด และสนับสนุนเฉพาะความสามารถในการสร้างแอพพลิเคชันและคอนโทรล ActiveX เท่านั้น

Visual Basic 5.0 รุ่น Professional

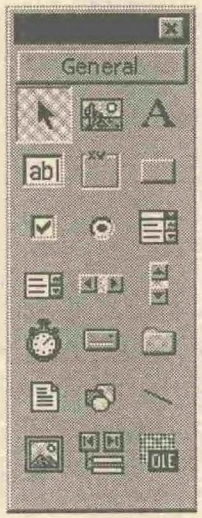
เป็นรุ่นที่มีความสามารถสมบูรณ์ที่สุด สำหรับการพัฒนาแอพพลิเคชันแบบ stand alone แต่ไม่สนับสนุนความสามารถด้านการพัฒนาแอพพลิเคชันแบบใช้ฐานข้อมูลร่วมกัน (share development) หรือผู้พัฒนาหลาย ๆ คน (multi-developers)

Visual Basic 5.0 รุ่น Enterprise

เป็นรุ่นที่มีความสามารถสมบูรณ์ที่สุด สำหรับการพัฒนาแอพพลิเคชันแบบองค์การ สนับสนุนความสามารถด้านการพัฒนาแอพพลิเคชันแบบใช้ฐานข้อมูลร่วมกัน (share development) หรือผู้พัฒนาหลาย ๆ คน (multi-developers) และการเข้าถึงฐานข้อมูลของระบบจัดการฐานข้อมูลที่มีชื่อเสียง เช่น Oracle หรือ Microsoft SQL Server เป็นต้น ด้วยเทคโนโลยี RDO อันทรงประสิทธิภาพ

ซึ่งสามารถที่จะทำการเปรียบเทียบคุณสมบัติและขีดความสามารถได้ดังในตารางที่ 1

สำหรับรายละเอียดทั้งหมดในบทความตอนนี้ ผู้เขียนได้คัดลอกมาจากเนื้อหาบางส่วนของบทนำ และบทที่ 1 ของหนังสือ การใช้โปรแกรม Visual Basic 5.0 Professional ซึ่งนอกเหนือจากเนื้อหาทั้งหมดนี้แล้ว ผู้อ่านสามารถที่จะศึกษารายละเอียดอื่น ๆ เช่น การแก้ไขปัญหาความไม่คอมแพทิเบิลระหว่างการติดตั้ง Visual Basic 4.0 และ Visual Basic 5.0



รูปที่ 8
แสดงหน้าต่างกล่องเครื่องมือ (ทูลบ็อกซ์)

การแก้ไขปัญหาฟอนต์ภาษาไทยกับเอนไวรอนเมนท์ของ Visual Basic 5.0 ที่ทำงานภายใต้ระบบปฏิบัติการวินโดวส์ 95 เอดิชันภาษาไทย การใช้คำสั่งทั้งหมดของเอนไวรอนเมนท์ของ Visual Basic 5.0 และคอนโทรล ActiveX ทั้งหมดของ Visual Basic 5.0 รุ่น Professional และอื่น ๆ อีกมากมาย ก็ขอให้ผู้อ่านที่สนใจ Visual Basic 5.0 คอยติดตามได้จากร้านหนังสือชั้นนำทั่วประเทศครับ

เทคนิคประจำคอลัมน์

ถาม คลาสมาตรฐานของวินโดวส์ 95 กับคอนโทรลภายในของ Visual Basic สัมพันธ์กันอย่างไร?

ตอบ วินโดวส์ 95 จะมีคลาสมาตรฐานที่ถูกสร้างไว้ภายใน ซึ่งถือได้ว่าเป็นคลาสร่วมมาตรฐานที่ทุก ๆ แอพพลิเคชันที่ทำงานภายใต้วินโดวส์ 95 สามารถกำหนดให้ตัวเองมีคุณสมบัติเดียวกับคลาสเหล่านั้นได้

Visual Basic ได้มีการสร้างคอนโทรลภายใน ที่มีการกำหนดคุณสมบัติให้ตรงกับคลาสมาตรฐานของวินโดวส์ 95 ซึ่งแต่ละคอนโทรลภายในจะตรงกับคลาสมาตรฐานของวินโดวส์ 95 ดังต่อไปนี้

System Control Class	Visual Basic Instrinsic Control
BUTTON	Check box, Option button, Command Button
COMBOBOX	Combo box : list always visible, drop-Down
EDIT	Text box : Single line, Multiple line, Password
LIST BOX	List box : Single selection, Multiple selection
SCROLLBOX	Scroll bar : Horizontal, Vertical

ตารางที่ 1

คุณสมบัติ	จุดเด่น	Enterprise	Professional	Control Creation
เอนไวรอนเมนท์การพัฒนาแอพพลิเคชันที่สามารถปรับปรุงตบแต่งได้	เอนไวรอนเมนท์ของ Visual Basic 5.0 ได้มีการเพิ่มเติมคุณสมบัติ IntelliSense ที่ช่วยในการเพิ่มความสามารถของเอนไวรอนเมนท์ เช่น QuickInfo, DataTips, List Properties/Methods หรือ CompleteWord เป็นต้น ซึ่งจะช่วยลดการจดจำคำสั่งและออบเจ็กต์ต่าง ๆ ได้อย่างมากมาย	☑	☑	☑
Visual Basic Applications Edition เวอร์ชัน 5.0	Microsoft Office 97 สนับสนุนการเขียนโค้ดเพื่อขยายขีดความสามารถหรือปรับแต่งการแสดงผลด้วยโค้ดของ Visual Basic 5.0	☑	☑	☑
โปรเจ็กต์ต้นร่าง และตำราคู่มือของ Visual Basic 5.0 แบบ online	ผู้อ่านสามารถสร้างแอพพลิเคชันได้จากโปรเจ็กต์ต้นร่าง ซึ่งจะช่วยลดความซ้ำซ้อนและเพิ่มความเป็นมาตรฐานในการออกแบบฟอร์มมากยิ่งขึ้น พร้อมทั้งตำราคู่มือของ Visual Basic 5.0 แบบ online ที่สามารถเรียกดูได้จากเอนไวรอนเมนท์ของ Visual Basic 5.0	☑	☑	☑
โปรแกรม add-in	สร้างและใช้งานโมดูลคลาส คุณสมบัติ โพรซีเดอร์ เหตุการณ์และวิธี พร้อมทั้งการออกแบบอินเทอร์เฟสของคอนโทรล ActiveX ได้โดยตรงจากเอดิเตอร์	☑	☑	☑
พัฒนาหลาย ๆ โปรเจ็กต์	เอนไวรอนเมนต์ของ Visual Basic 5.0 สนับสนุนการพัฒนาและดีบักโค้ดแบบหลายๆโปรเจ็กต์ พร้อมทั้งสามารถรันหลาย ๆ โปรเจ็กต์ในเวลาเดียวกันอีกด้วย	☑	☑	☑
การสร้างคอนโทรล ActiveX	สนับสนุนการสร้างคอนโทรล ActiveX ที่สามารถใช้งานร่วมกับ Visual Basic 5.0, Visual C++, Microsoft Office หรือ Web applications	☑	☑	☑
การสร้างแอพพลิเคชัน	สนับสนุนการสร้างไฟล์ .exe และ .dll (OLE DLL)	☑	☑	
การเข้าถึงไฟล์ฐานข้อมูล	การเข้าถึงไฟล์ฐานข้อมูลด้วย DAO หรือโดยอาศัย ODBC เพื่อสมรรถนะที่เหนือกว่า	☑	☑	
คอมไพเลอร์ที่สามารถออปติไมซ์โค้ดแท้	คอมไพเลอร์ของ Visual Basic 5.0 สามารถที่จะสร้างโค้ดแท้ที่ผ่านการออปติไมซ์ เช่นเดียวกับคอมไพเลอร์ของ Microsoft Visual C++	☑	☑	
Active Document	สนับสนุนการสร้างแอพพลิเคชันชนิดอ้างอิง browser ที่เรียกกันว่า Active Document ซึ่งสามารถเรียกดูได้โดยเบราเซอร์ Microsoft Internet Explorer	☑	☑	
Microsoft Transaction Server Developer Edition	สนับสนุนการสร้างแอพพลิเคชันด้านฐานข้อมูลแบบสถาปัตยกรรม three-tiered component	☑		
สนับสนุน Native RISC	สนับสนุนการสร้างแอพพลิเคชันที่ทำงานภายใต้สถาปัตยกรรม RISC อย่างสมบูรณ์	☑		
Remote Data Object (RDO) เวอร์ชัน 2.0	สนับสนุน RDO สำหรับการเข้าถึงฐานข้อมูล ODBC แบบความเร็วสูงจากอินเทอร์เน็ตหรือแอพพลิเคชันแบบไคลเอ็นต์เซอร์ฟเวอร์	☑		
TSQL Debugger	ดีบัก SQL Server stored procedures และ triggers อย่างโต้ตอบในเวลาเดียวกัน	☑		